COGNITIVE AUGMENTATION FOR NETWORK DEFENSE

THESIS

James E. Emge, Captain, USAF

AFIT-ENG-13-M-16

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENG-13-M-16

COGNITIVE AUGMENTATION FOR NETWORK DEFENSE

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Cyber Operations

James E. Emge, B.S.C.E.

Captain, USAF

March 2013

AFIT-ENG-13-M-16

COGNITIVE AUGMENTATION FOR NETWORK DEFENSE

James E. Emge, B.S.C.E.
Captain, USAF

Approved:

Kenneth Hopkinson, PhD (Chairman)	Date
Timothy Lacey, PhD (Member)	Date
Robert Mills, PhD (Member)	Date
Maj. Kennard Laviers, PhD (Member)	Date

AFIT-ENG-13-M-16

Abstract

Traditionally, when a task is considered for automation it is a binary decision, either the task was completely automated or it remains manual. Level of Automation (LOA) is a departure from the tradition use of automation in cyber defense. When a task is automated, it removes the human administrator from the performance of the task, compromising their Situational Awareness (SA) of the state of the network. When the administrator loses SA of the network performance and its current state, failure recovery time becomes much longer. This is because the administrators must orient themselves to the current state of the network at the time of failure and determine the cause of the failure before repairs or supplemental operations can occur. LOA attempts to mitigate this problem by keeping the administrator engaged in network tasks along side the automation agent. Keeping the administrator aware of both the automated system's performance and the performance of the network, while taking advantage of the automation system's speed and the complex decision making of the administrator. This research applies LOA to computer network defense during cyber attacks. The goal is to find the most efficient LOA that keeps the administrator engaged in the defense of the network while preserving efficiency. The LOA allows the administrator to supplement and/or correct the automated system, while the automated system handles the time sensitive events to keep the administrator from being overwhelmed or the network from being compromised.

This Theis is dedicated to my wife who endured my absence with loving patience and to my lovely daughter, who made going home a delight.

Table of Contents

Page

List of Figures

List of Tables

List of Acronyms

Acronym	Definition
AD	Active Directory
BASE	Basic Analysis and Security Engine
CUT	Component Under Test
DOS	Denial of Service
ES	Expert System
GUI	Graphical User Interface
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
LOA	Level of Automation
SA	Situational Awareness
SUT	System Under Test
UAV	Unmanned Aerial Vehicle
MNDI	Mobile Network Defense Interface

COGNITIVE AUGMENTATION FOR NETWORK DEFENSE

I. Introduction

Network administrators have been trying to find a balance between security and usability since computer networks were used in business transactions. In the past, security was primarily maintained by network policies and configurations [7]. The network administrator maintained Situational Awareness (SA) on their network because they were actively engaged in network operations. Currently, automated systems monitor a vast majority of network operations and administrators monitor the automated systems, removing them from directly interacting with network functions and reducing their SA of the current network state [13][18][24].

The amount of information that computer networks produce is overwhelming for network administrators. From the amount of information transmitted across the network to the vast array of diagnostic information contained within the nodes of the network, there is too much to sort through manually [7]. Information rapidly increases during periods of elevated cyber attacks and information overload for the network administrator is quickly reached at a time when human intervention is required most [14]. To help the network administrator manage all of this information, automated systems monitor, parse, record, act on, and present information to the administrator. These systems include Intrusion Detection Systems (IDSs), firewalls, routers, packet scanners, among others [28]. Automation has created a time gap between an event occurring and the administrators response to the alert. With the speed networks operate, a time gap of a few minutes between alert and resolution can cripple network operations ranging from minutes to days depending on recovery time.

An approach to closing this gap is to find a more effective way to include administrator interaction with the automated system. This research focuses on determining the most efficient balance of control between automated network systems and network administrators.

Level of Automation (LOA) is a departure from the traditional use of automation. Traditionally, when a task was being considered for automation the decision was binary, either the task was fully automated or it remained manual. When the task is fully automated it removes the human administrator from directly interacting with the operation of the task which creates the out-of-the-loop performance issue [19][18]. LOA is not a new concept and has been applied to power plants, information gathering, and Unmanned Aerial Vehicle (UAV) control [13][32][23]. This research focuses on how adjusting the LOA impacts the network administrator's performance.

The research goal is to determine what LOA would best preserve a network administrator's SA and performance during cyber attacks. These experiments will look at administrator performance and SA as the LOA, workload, and interface platform are varied. The research will show what LOA best divides the workload among the administrator and Expert System (ES) and if that setting is interface independent.

II. Literature Review

LOA has traditionally been applied to applications such as aviation, robotics, information processing, and controlling multiple UAVs in flight [13]. This research applies LOA and moves the administrator back into direct operation of the network with the goal of increasing network SA.

2.1 Situational Awarness

SA is a subjective metric, biased on a person's awareness of the environment. Extensive research has been conducted on how to measure SA. Dr. Endsley introduced the fundamental model of SA containing three levels; perception, comprehension, and projection [12]. Level 1, perception is the gathering of information about the environment and the variables that can effect its state. Without a well formed and accurate perception of the environment the other levels will be flawed and inaccurate. Level 2, comprehension is the sorting, combining, processing, and interpretation of the information obtained during perception. During this level a cognitive model of the current state of the environment is developed by the subject. For this model to be accurate it must include all the important factors discovered during perception. There is a constant flow of information from the perception level that allows the subject to update their model state at the comprehension level. This model is used to predict the environment's future state; which is level 3, projection. A 4th level is purposed by McGuinness and Foy called resolution [21]. Resolution builds on level 3 by trying to find the optimal path to achieve a desired state change. This model for representing SA provides a well defined structure for measuring SA during this research and can be applied to both an human or automated system [17].

2.2 Manual Network Defense

2.2.1 Configuration and Policies.

Before any active defense or monitoring systems are used, configuration and policies are deployed to secure and harden networks against cyber threats [7][9][16]. Configurations are categorized in two major areas, physical and software. Physical configurations can limit how many physical paths exist between the local network and others including connection to other networks. This simplifies the later deployment of additional network security systems. Proper software configuration can help further secure the network by controlling what types of service requests the network will respond to. An example of this are firewall configurations that stop ping, Secure Shell (SSH), Hypertext Transfer Protocol (HTTP), and File Transfer Protocol (FTP) requests to portions of your network that are not designated to respond to these requests and thereby eliminating a possible attack vector.

Network policies are instructions that outline human interaction with the network and are used to keep network users and operators from performing actions that circumvent network security [9]. Examples of common network policies are password requirements to prevent easily broken passwords or a rule forbidding network users from using personal storage devices to mitigate the spread of viruses.

Configurations and policies are necessary to secure a network but are not sufficient by themselves to secure todays networks [9]. Policies and configurations are labor intensive to change and cannot be adapted fast enough to counter a dynamic cyber attacks.

2.2.2 Monitoring.

Manual network defense is rarely used to counter cyber attacks in real time. The network administrator would not know the network is under attack until some systems start malfunctioning or an anomaly is noticed during a history log review [6].

At this point, the attack could have already been executed, and the administrator has to recover and try to close the discovered attack vector. The benefit of having the administrator directly interacting with the network is that they are already aware of the network state before the attack happens. This could make recovery time faster [18]. However, it would be more efficient if the network had the ability to detect intruders and defend itself and the administrator could keep SA at the same time.

2.3 Automated Network Defense

Automated systems can analyze more data faster and react much quicker than a human administrator [9]. Network attackers utilize their own automated systems to rapidly execute multiple attacks and move through the target network and cover their tracks. Network defenders are forced to counter this capability by installing automated systems as well. The result is the administrators primary responsibility has shifted from network operations to monitoring the automated system.

2.3.1 Intrusion Detection Systems.

IDSs are used to monitor network traffic and look for patterns that might indicate malicious intent [9]. When the IDS detects a possible network attack, it generates an alert that contains the pertinent information on the network intrusion. That alert can be sent to a number of other systems, like an email account or other automated alert system, or it can be logged in a history file to be reviewed by the administrator.

A limiting factor to an IDS's ability to detect attacks are the two main methods used to analyze network traffic; anomaly and misuse detection [9]. Anomaly detection compares known good logs of acceptable activities on the network to monitored traffic. Administrator can defined rule sets to configure the IDS for custom activities that are authorized or unauthorized on the network [9]. Misuse detection, or pattern matching, uses static signature files, much like anti-virus programs, to detect common

traffic patterns seen in network attacks. A limitation of this method is if a new attack occurs that is not in the signature file, or if an attack is made to look like normal network traffic, it can escape detection [9]. Encrypted traffic is another problem for an IDS. IDSs work as packet sniffers so when network traffic is encrypted, the IDS is not capable of examining the data in the traffic without the ability to decrypt the traffic. The process of decryption is a resource intensive process and can violate network security protocols [9]. An additional shortfall of the IDS is that it only detects possible attacks, there is no mechanism for the IDS to take action on the network to subvert the attack [9].

An IDS is primarily a forensic or analysis tool rather than a defense system [9]. The problem is most attacks happen too fast for a human to respond [7]. IDS alerts are used by the network administrators to track attacks that have already happened, focusing their efforts in places where intruders are most likely hiding and to identify where the network is most vulnerable. It is much more efficient than scanning multiple log files but still not the a timely enough response needed to defend a network from today's attackers.

2.3.2 Intrusion Prevention Systems.

Intrusion Prevention Systems (IPSs) are an extension of an IDS. They utilize the same detection methods as an IDS and are susceptible to the same shortfalls of those methods. The key difference is that IPSs are capable of taking action to change the network to counter a network attack [9]. This new ability does not come without its own deficiencies. An IPS can only act on network nodes that it has been integrated with. IPSs are commonly integrated with other network defense systems, like firewalls, but are less commonly integrated workstations or hardware with uncommon operating systems, introducing a vulnerability.

2.3.3 Anti-Virus and Internet Security.

Both IDSs and IPSs focus on network traffic analysis and do not account for the social engineering aspect of cyber attacks [20]. If a network users gets tricked into installing a foreign program or into revealing their password, an attacker using their credentials could bypass an IDS or IPS and install malicious code on the network. This is where anti-virus and Internet security software come into effect [7]. These software programs scan local hosts for malicious code and remove or quarantine infected files that could allow remote access or data exfiltration from network assets.

2.3.4 Other Automated Network Defenses.

The one common failing automated network analysis and defense systems is that they are not immune to error or failure. These systems are monitoring and reporting on the status of the network but there are no systems monitoring the monitors, this is traditional the administrators' job. The human administrators then shifts their focus to administrating and monitoring the automated systems instead of the operational network [6]. These automated systems serve a vital function but the complete removal of the administrator from the operational network degrades their ability to be effective when the automation system fails.

2.4 Level of Automation and Human in the Loop

LOA is the introduction of automation to assist a human administrator in a task rather than replacing them [25]. This can be a delicate balance because with too much automation, the administrator's task becomes that of monitoring the automated system, which introduces the out-of-the-loop performance issue [19][11]. Automation bias also becomes an issue when human administrators interact with automated systems [10][29]. Automation bias sets in when a human operator stops relying on their own knowledge and starts to defer decisions and enacting recommendations made by an automated system with out cross checking the validity of those actions.

With too little automation and the operator can quickly become overwhelmed. Both conditions can have major drawbacks.

2.5 Human Vigilance and the Complacency Problems

Introducing automation in a system and relegating administrators to monitoring automated systems for infrequent changes gives rise to the vigilance and complacency problem [31]. Human administrators are not mentally equipped for vigilance tasks [31]. The administrator will become complacent and assume that the system is operating as designed and that leads to automation bias where the administrator defers decisions to the automation system without checking to see if the actions is the best course of action [22][8][26].

2.5.1 Full Automated Control : Monitoring.

In a fully automated network, the administrator is relegated to monitoring the automated systems. As the operator interacts with the automated systems, they can monitor status, alerts generated and actions taken on the network. The automated system's ability to react faster than a human operator gives it some powerful advantages over a manually defended network [30]. As long as the automated systems function correctly, the operator has high confidence that the network is running as desired. Problems arise when the automated system fails and the operator has to manage the network manually. Skill degradation can make the administrator ineffective at manual control of network function [10][11]. Automation bias can degrade an administrator's ability to decide the best course of action in an event where there is no automated recommendation. Administrator complacency can lead to a undetected failure if the automated system fails to register it [19].

Research into human vigilance has shown that humans are ill suited to tasks like monitoring a system for infrequent changes [31]. Automation can alleviate the burden on the human administrator, however if the administrator just shifts what system they

monitor the same issues are still present. The further removed the administrator is from the decisions that affect the operation of the network the more likely automation bias will occur [10][6]. Automation bias occurs when an administrator fails to undo a automated system's actions even in the event of information that is contrary to the automated systems actions or recommendations. Automation bias can lead to failures from the operator's lack of vigilance in checking the automated system's actions and failing to recognize aberrant behavior because of unexpected behavior from the automated system [29]. A classic example of this is a server failure and an automated system that activates a backup server. The administrator is complacent and trusts the automated system, the failure could go unnoticed until the backup server fails as well.

When the automated system fails and forces the operator to manage the network manually, out-of-the-loop performance becomes an issue along with skill degradation [25][18]. The out-of-the-loop performance issues arise from the operator's separation from the network's current state. The operator is slower to respond to requests and alerts because they have to spend time familiarizing themselves with the current network state before they can act efficiently on the network. Skill degradation comes from their lack of practice with manual commands. The operator might forget critical commands during a time sensitive operation or fail to run all the proper actions due to their extended time not directly interacting with network.

2.5.2 Full Manual Control.

The extreme is full manual control of the network. This is not commonly used in today's enterprise level networks because of their size, scope, and complexity. However, it can be implemented in a limited scope, because of legacy systems or new systems that cannot integrate with existing automated systems. As the operator becomes efficient with the new systems, their ability to administrate and deal with

unexpected errors will improve. The benefit the operator gains through the direct interaction with the system is they are familiar with the system's operation, bugs, and what normal operation looks like on a day-to-day basis. This, potentially, shortens the time it takes to diagnose a problem and implement a fix [11][19]. The downside is that as the complexity and/or information flow increase in a manually maintained system it increases the chances that an operator will get overloaded, thus decreasing their efficiency [30].

2.5.3 Partial Administrator Control: Levels of Automation.

Partial administrator control or LOA attempts to combine the benefit of automation and administrator control. LOA has been applied to the nuclear power field with the goal of increasing SA and reducing the out-of-the-loop performance issues [13][19][11]. In a nuclear power plant, it is important to identify, track, and correct problems quickly to ensure safe operations and continuous service. LOA has the potential to provide the same benefit for cyber defense. Both types of systems require operators to know system specific commands and/or actions to manually operate the system as well as operators and administrators are required to monitor the systems for infrequent status changes.

By keeping operators engaged in the operation of the network, their efficiency will increase. Allowing the operator to engage and use commands directly on network systems achieves this goal. Thus reducing the operator out-of-the-loop performance issue and skill degradation. It can also reduce the human vigilance issue by breaking up monitoring (vigilance) tasks with active tasks, and lead to an increased SA for administrators.

As the future state of warfare relies heavily on cyber assets and our enemies and allies continue to gain experience and new technologies in this field. We need to continue to develop new and better methods to defend our critical assets. As we

continue to employ new technology throughout the country, our computer networks are becoming a gateway to all of our vital infrastructure and have to be defended [5].

III. Methodology

3.1 Research Question

There is little in-line interaction of automated systems and human administrators. Because of the this separation between the human administrator and the active network nodes, issues like automation bias, skill degradation, and out-of-the-loop performance become a problem. This research presents a method to allow automated systems and human administrators to interact side-by-side on the network to help alleviate these issues.

3.2 Approach

The approach of this research consists of using volunteer network administrators that are evaluated on their abilities to defend a network in a controlled environment. Using, either, a desktop configuration or a Mobile Network Defense Interface (MNDI) loaded with network interface software. The administrators will manage a network during four test cases, each at a different LOA. Each test case consists of four randomly generated cyber attack scenarios.

3.2.1 Network Interface.

3.2.1.1 Desktop.

The desktop interface consists of Spiceworks (Figure 3.1) which is a host monitoring tool deigned to track the status of network assets [4]. The main source of information presented to the administrator is an inventory of the test network assets, related IP addresses, and functions of the servers. It also presents a log of actions taken on the hosts and an up/down status.

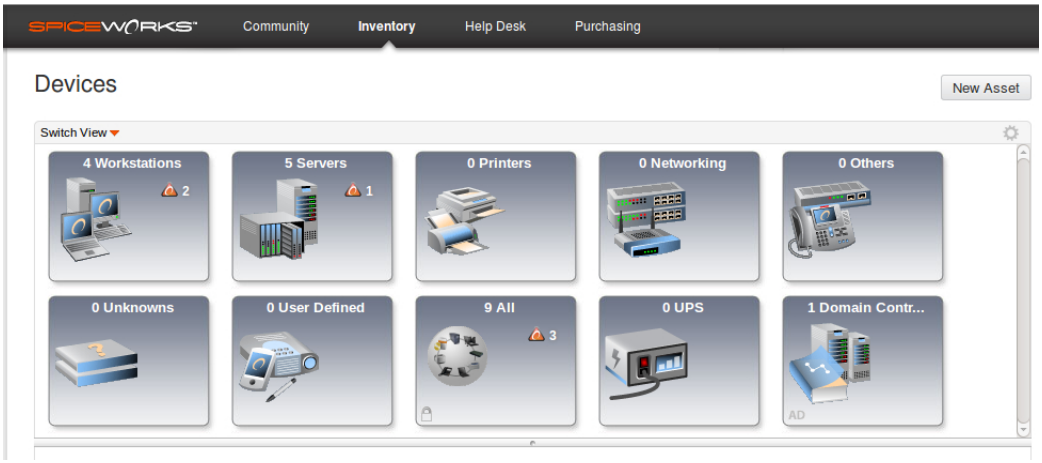

Figure 3.1: Spiceworks - Host Monitoring

SNORT [3] is the IDS that the administrator uses for network traffic analysis. With Basic Analysis and Security Engine (BASE) (Figure 3.2) [1] as the Graphical User Interface (GUI) for SNORT. This presents the administrator with alerts that signal possible network attacks. Each attack is assigned a unique ID that will be used to correlate user resolutions to a specific alert.

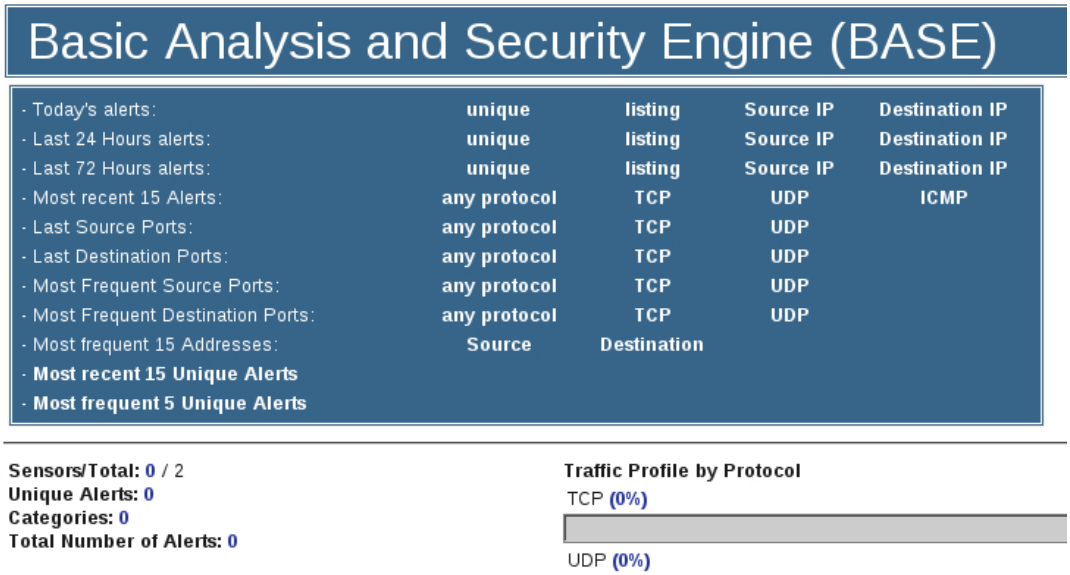

Figure 3.2: Basic Analysis and Security Engine (BASE)

The network interaction window (Figure 3.3) is a custom application that was written to give the administrator an interface to enter network action scripts and SNORT alert IDs to resolve alerts and allow logging of all administrator actions.

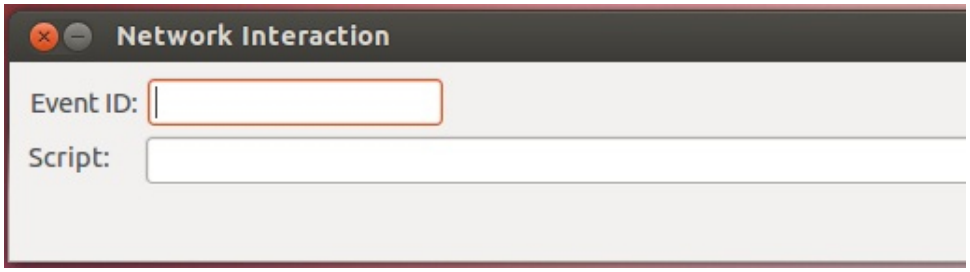

Figure 3.3: Network Interaction Window

The alert resolution window (Figure 3.4) is how the administrator knows what alerts have been resolved on the network. As the administrator and the automated

system resolve the alerts from SNORT, this windows is updated with that information.

Event ID	Signature Name	Source IP	Dest IP	Impact
45689	Possible TCP SYN Flood DoS	192.168.251.150	192.168.1.37	5
45693	Possible TCP SYN Flood DoS	192.168.1.180	192.168.1.20	4
45697	SQL injection attempt	192.168.1.133	192.168.1.37	5
45701	Possible TCP SYN Flood DoS	192.168.1.157	192.168.1.10	4
45705	Possible TCP SYN Flood DoS	192.168.1.159	192.168.1.10	4
45709	ET SCAN Potential FTP Brute-Force attempt	192.168.1.37	192.168.251.174	5
45717	POP3 Mailbox overflow attempt	192.168.251.192	192.168.1.5	3
45721	Possible TCP SYN Flood DoS	204.109.190.191	192.168.1.37	5

Figure 3.4: Alert Resolution Window

3.2.1.2 Mobile Network Defense Interface.

The mobile network controller is an iOS application written to interface with the test network. It provides administrators with a GUI to interact directly with the network and its automated systems [15]. The MNDI's information windows consist of network topology [Figure 3.5:#1], node health and link saturation information [Figure 3.5:#2], action history log [Figure 3.5:#3], and the action/alert window [Figure 3.5:#4]. The network topology displays the current network connections with a visual indication of bandwidth saturation. Each node's visual representation indicates if it is a server, firewall or workstation and if it is currently up or down. Tapping on each node shows an informational window which displays the health of the node. The window contains the IP address, operating system, CPU utilization, RAM, RAM unitized, network saturation, and packets lost rate of the selected node. The action history log is a list of alerts, administrator actions, automated actions, and resolved alerts presented in a color coded list to the administrator. It contains the unique alert ID, source and destination IP addresses

for the actions and alerts as well as a time stamp. The alert window contains the alert ID, type of alert, source and destination IP addresses, and network impact rating. The impact rating is a integer between 1 and 7, with one 1 representing the lowest impact and seven 7 as the highest [Table 3.1].

The administrator interacts with the network through two separate windows, the action and alert windows. Each network node has an associated action window that list all available actions for the selected node. Once the administrator chooses an action the appropriate script is generated and submitted for action. Th alert window is displayed when a new network alert is generated. This window has the relevant information about the alert as well as lists the appropriate actions for resolving the alert. The ES's suggested action is also highlighted in the window.

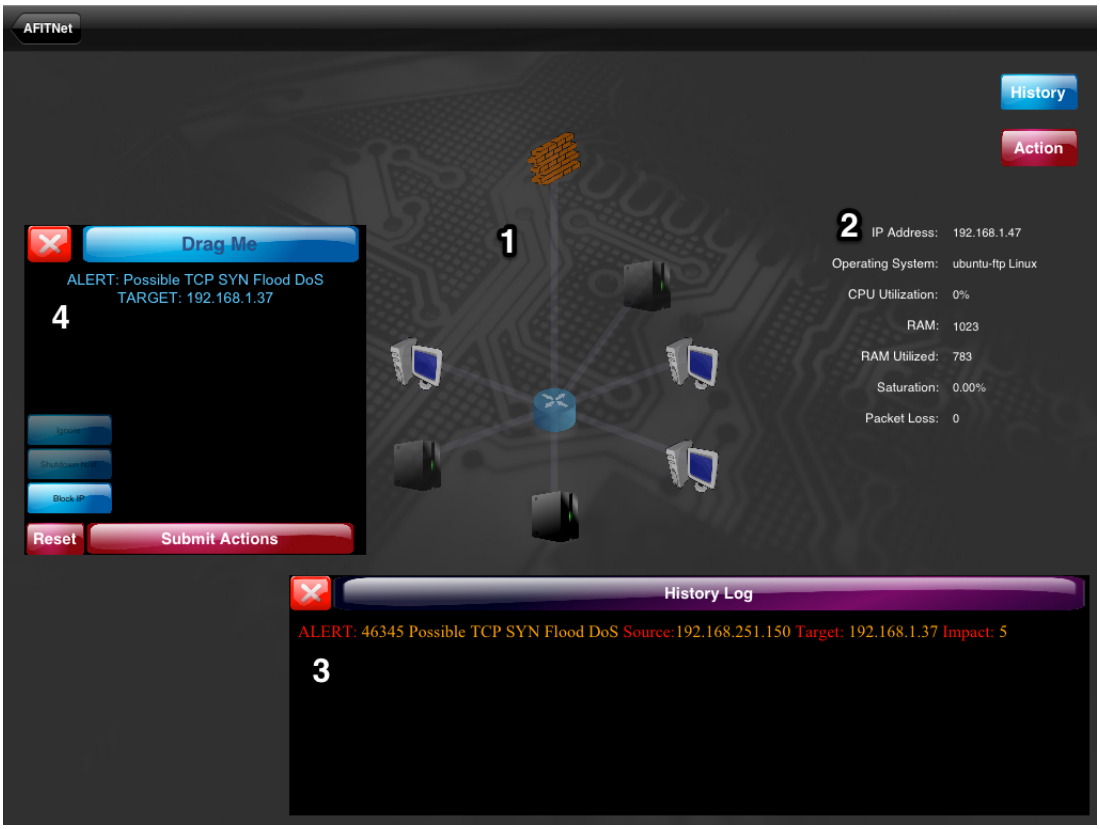

Figure 3.5: MNDI Interface

Table 3.1: Impact Levels [27]

Impact	**Description**
7	Target host is vulnerable and attack results in high data loss or service degradation
6	Target network contains hosts that are vulnerable and attack results in high data loss or service degradation
5	Target host is vulnerable and attack results in moderate data loss or service degradation
4	Target network contains hosts that are vulnerable and attack results in moderate data loss or service degradation
3	Target host is vulnerable and attack results in minor data loss or service degradation
2	Target network contains hosts that are vulnerable and attack results in minor data loss or service degradation
1	Target network is not vulnerable to attack vector

3.2.2 Automated System : Expert System.

The ES is integrated into the MNDI. Its decisions are based on a truth table which is generated based on the cyber attack types that are used in the research.

The ES consists of:

- Alert monitoring processes that tracks current network alerts and generates scripts to changes the network configuration based on the truth table.
- Level of Automation (LOA) system which will automate alert resolution based upon the LOA setting and the alert's impact on the network.

- Script execution system.

The design and capabilities of the ES are not under study. This portion is built from existing systems and research. The goal is to evaluate the most effective level of assistance a ES system should provide, independent of the actual ES system. For this research the ES reads in the list of alerts from the network modeler, discussed in Section 3.2.3, and loads a list of actions that will resolve the alert from a truth table. The ES then automates the response or generates a recommended actions biased on the impact level of the alert and a threshold value given to the ES.

3.2.3 Network Modeler : Data Fusion.

Both the MNDI and the ES draw their data from a network modeler installed in the test network [27]. This network modeler pulls its network picture and service list from network scans performed by PBNJ [2]. The node health information is gathered by a Java application that was installed on the network hosts [27]. The network alerts are pulled from SNORT [3] and stored in the model under the associated node. Each alert generated by SNORT is analyzed by the network modeler and assigned an impact level based on the type of attack, if the target of the attack is vulnerable, and if there are any nodes vulnerable to the attack. This impact level determines if the ES automates the resolution or if it is presented to the administrator.

3.2.4 Test Network Configuration.

The test network, as shown in Figure 3.6, consists of four workstations, six servers and the desktop and MNDI. The administrator will interact with the three windows workstations, pfSense firewall, the Ubuntu web and FTP servers, and the Microsoft Active Directory (AD) and Exchange server. The Spiceworks and network modeler servers are responsible for supplying the network status to the administrator and the BackTrack 5 workstation is running the software that generates cyber attack test cases for the network.

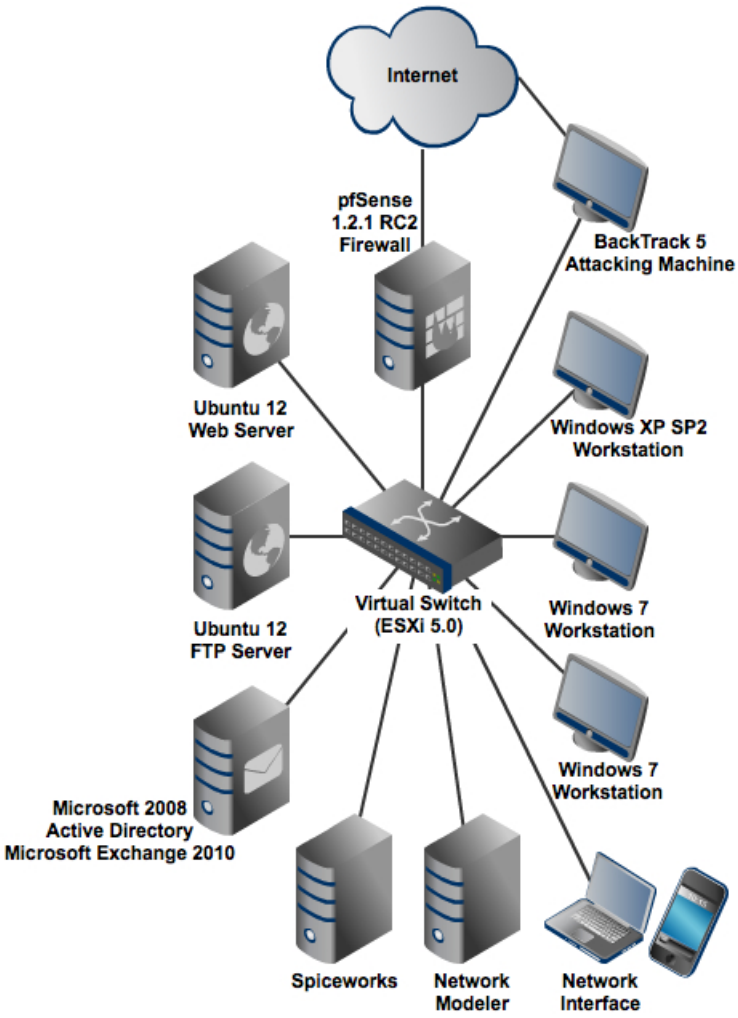

Figure 3.6: Test Network Diagram

3.2.5 Recoverability.

Once the test environment is configured, it is used in the same configuration for all of the experiments performed. The network is built inside of a virtual environment which allows the entire network to be reverted to a previous state. An automated cyber attack generator is installed into the network to give the automated systems and network administrators traffic and attacks to monitor and adapt to.

3.3 System boundaries

The System Under Test (SUT) is the interaction of the LOA contained in the ES and the network administrator, as shown in Figure 3.7. The network interface is chosen randomly for each test subject. The type of interface is not under test but both data sets will be examined to determine if the optimal LOA is dependent or independent on interface type.

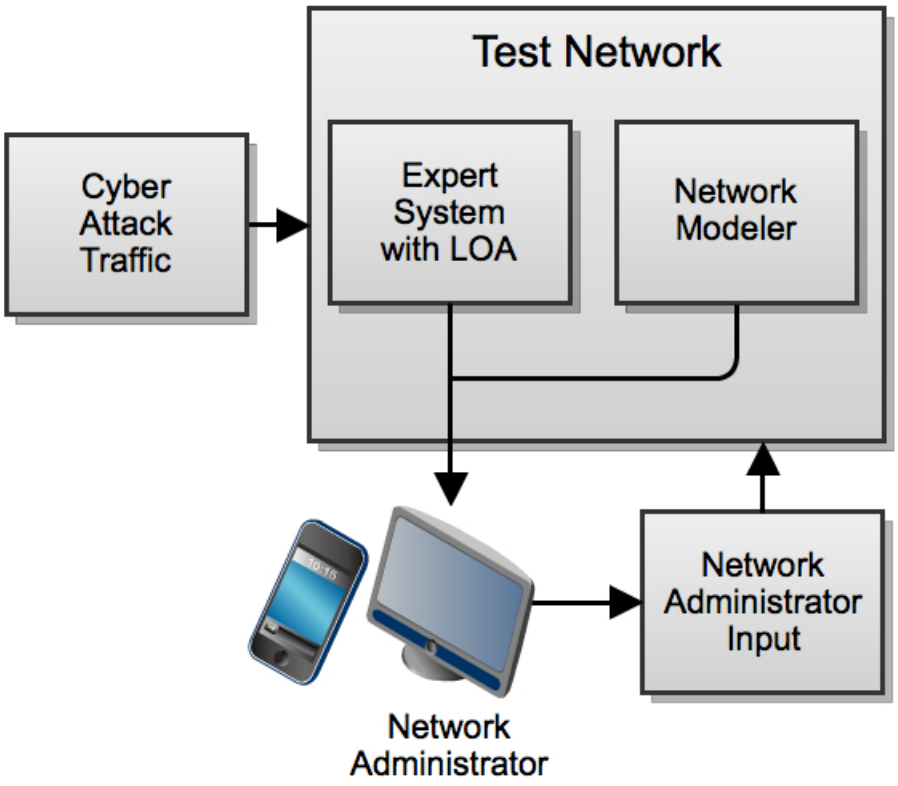

Figure 3.7: Network Attack Information Flow

The ES is responsible for monitoring and sorting incoming network alerts and automating the resolution for the alerts that meet the impact threshold. The administrator is then presented with the alerts that fall below the LOA impact

threshold (MNDI only) and with a list of alerts that were auto resolved (MNDI and desktop interface). The Component Under Test (CUT) is the LOA in the ES which controls the impact level threshold used to determine which alerts are automated.

The main limiting factors in this study are the size and complexity of the test network and the number of test subjects able to be tested in the time allotted. To keep the implementation of the network interfaces and the ES simple enough to be completed in the time allotted simplistic cyber attacks and network configurations are used. This could affect how the results translate to larger networks with more elaborate automated systems and a broader scope of cyber attacks. The configuration of the network is illustrated in Figure 3.6.

3.4 System Services

The ES sorts the alerts provided by the network modeler into two list; automated response and user response. The automated response list is processed by the ES and the appropriate response is taken. The user response list is presented to the administrator with a recommendation provided by the ES (MNDI only). [Figure 3.8]

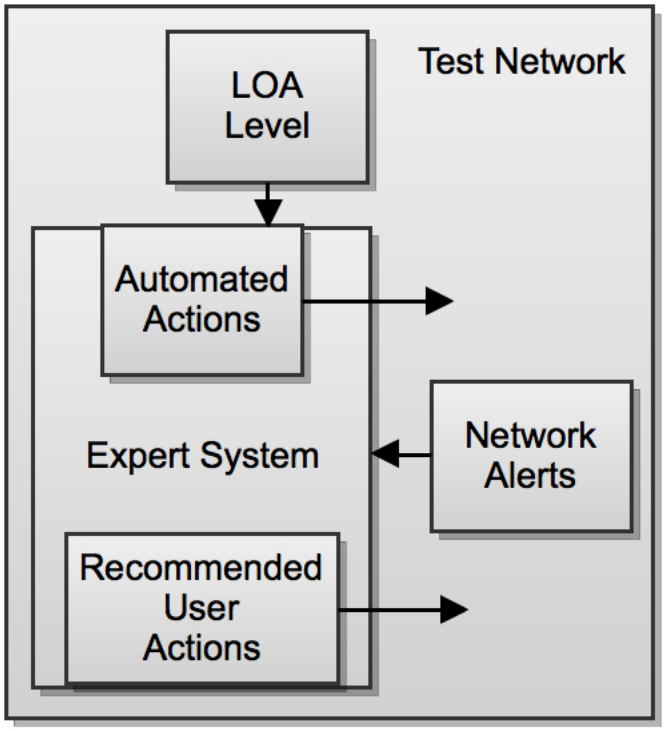

Figure 3.8: Expert System (ES) Flow

3.5 Workload

The workload for the network consists of cyber attack scenarios among a background of normal network traffic inherent in the test network. A test case is run at each level of LOA. A test case is made up of four scenarios separated by a randomly generated delay between 120,000 and 300,000 ms. Each scenario consists of a random number of attacks between seven and twenty-five, and each attack is delayed from the previous one by 0 to 10,000 ms. The test cases provide the ES and the administrator alerts to respond to. The alerts are deliberately not covert and do not try to evade the automated detection systems. Detection times are not considered; only the time from alert to resolution is considered.

Table 3.2: Workload Parameters

Parameter
Delay Time between Scenarios
Number of Cyber Attacks in each Scenario
Delay Time between Cyber Attacks

3.6 Performance Metrics

The first two performance metrics are percentage of the alerts resolved and percentage of user resolved alerts. These metrics show the effectiveness of the overall system and if it can handle the supplied workload. The metrics used to determine SA are the number of the administrator's additional actions and percentage of user alerts resolved. These metrics show the administrator's involvement in the network and demonstrated if the administrator was overwhelmed or not.

Table 3.3: Metrics

Metric
Percentage of Total Alerts Resolved
Percentage of User Alerts Resolved
Number of Administrator Actions

3.7 System Parameters

The following parameters can affect performance of the SUT.

- Types of Network Traffic: Certain types or combination of types of traffic could give false positives or negatives as the IDS monitors for attacks.

- Network Modeler: Different Network Modelers could interface differently with the ES. Performance could be affected depending on the amount of processing the ES has to do on the data provided.

- Types of IDSs: Different IDSs interfaces differently with the network modeler. Performance could be affected depending on the amount of processing the network modeler has to do on the data provided.

- Type of Network Interface: The administrator is randomly assigned to either the desktop interface or the MNDI. The type of interface affects administrator performance.

- Level of Automation: This is set to one of four levels and is the factor in this experiment. The levels are 1, 5, 6, and 8.

- Number of Cyber Attacks per Scenario: The number of attacks must be high enough to get a distinct difference, if one exists, between the different assistance levels and administrator performance. However, not so high as to compromise the normal function of the ES.

- Delay between Scenarios per Experiment: How much time the administrator and the ES get to resolve the current alerts before the next set come in.

3.8 Factors

The factors in this experiment are the level of assistance provided by the ES, Network Traffic Density, Number of Simultaneous Scenarios, and Level of Network Operator Knowledge. Table 3.4 lists the factor levels and they are defined below.

Table 3.4: Factor Levels

Factor	**Level**
Levels of Assistant	1, 5, 6, 8
Number of Cyber Attacks per Scenario	7-25
Type of Network Interface	Desktop or MNDI
Level of Network Administrator Knowledge	Varies between person. All volunteers fill out a self assessment questionnaire

3.8.1 Levels of Automation.

The LOA are defined as follows: 1, 5, 6, 8. These are compared against the alert's impact level. At level 1 all alerts with impact 1 and higher are automated. This means that the ES automates all the changes to the network in response to an alert and then presents those decisions to the administrator. The administrator can counter those changes or supplement the ES's actions. At level 5 the ES will automate approximately 75% of the network alerts, and the administrator is required to respond to the others. The administrator is still free to change or supplement the ES's actions. At level 6 the administrator is required to handle the majority of the alerts and the ES only handles alerts with an impact of 7. Level 8 is fully automated with no automated actions.

A difference aries between the desktop and MNDI when interacting with the ES at levels 5, 6, and 8. The MNDI has access to the ES recommend actions for each alert that requires administrator action, where the desktop interface does not.

3.8.2 Number of Cyber Attacks.

Number of Cyber Attacks per Scenario: This factor varies randomly from seven to twenty-five, in order to measure the differences in the human-machine interaction at different workload levels.

3.8.3 Type of Network Interface.

This factor varies to determine if the response variable is dependent on the type of interface.

3.9 Evaluation Technique

Simulation is used to evaluate system performance. This is the best way to control the other parameters values, prevent undesired fluctuations, and to make sure the experiment and the results are repeatable. In a real world measurement it is infeasible to control or account for the variations in network traffic that would occur during the experiments. The simulation is configured as follows. The network is a self contained virtual environment housed in a ESXi 5 server. The network is built around a Microsoft 2008 active directory structure with Exchange 2010, web services, and workstation machines. Once the network is configured, snapshots of all the servers are taken and the entire network is cloned to ensure it can be restored to its original state before each run.

3.10 Experimental Design

The network is composed of virtual machines running Windows 7, Windows XP SP2, Windows Server 2008, BackTrack 5, and Ubuntu 12 with Active Directory, Microsoft Exchange, Network Modeler, IDS, Web, and FTP network services. The cyber attacks vary randomly in order, number, target IP, and destination IP. This experiment is a classic block design with the network administrator and interface type as the blocking factors.

This experiment will be run as a within-subject study in a full-factorial design. There will be nineteen administrators, each participating in four runs of the experiment, one at each level of automated assistance.

3.11 Expriment Time Table

Each test subject who participates in this research is guided through the procedure according to the following time table [Table 3.5]

3.12 Methodology Summary

The test network is a closed system with a limited number of nodes and attack scenarios. The ES receives input from the network modeler and makes network changes based on its truth table to eliminate the attack vectors. The network administrator has a set of visualizations and controls to interact with the ES and make changes to the network. Each of the nineteen administrators perform the experiment four times, each at a different assistance level. All of the attack scenarios between runs and administrators are randomly generated. The data collected is used to determine what level of automated assistance and human interaction maximizes SA during cyber attacks and if the LOA is platform dependent.

Table 3.5: Expriment Time Table

Stage	**Description**	**Time in Stage**
Computer Network Proficiency Assessment	A self assessment questioner used to determine administrator network experience	5 min
Experiment Overview	An overview about why the experiment is being conducted what it is trying to achieve.	5 min
Training	This is the official training that the administrator receives about the environment and the network interface they are assigned.	10 min
Test Case 1	Run at a random LOA setting (1, 5, 6, 8)	20 min
Break		5 min
Test Case 2	Run at a random LOA, excluding the LOA from Test Case 1	20 min
Break		5 min
Test Case 3	Run at a random LOA, excluding the LOA from Test Case 1 and 2	20 min
Break		5 min
Test Case 4	Run at a random LOA, excluding the LOA from Test Case 1-3	20 min
Network Tool Assessment		5 min
Total		120 min

IV. Analysis of Results

This chapter discusses the distribution of automated alerts versus administrator alerts. Additionally, it supplies evidence that the workload used in this study was sufficient to overwhelm the administrator at the lower LOAs on both platforms. Next it looks at the efficiency of the administrator-ES system on both platforms. Specifically how does the efficiency change between the different LOAs and platforms. Finally, it will look at how we can determine the administrator's SA of the network state.

4.1 Distribution of Work at LOAs

At LOA 1, 100% of the workload is placed on the ES. The administrator's only required task is to monitor the ES and report any errors that occur. The administrator can execute any network actions they wish, but none are required to resolve alerts. The administrator main task type is a vigilance task during the LOA 1 test case. Human operators are not suited for this task, even at a short duration of twenty minutes. It was observed that during the LOA 1 test case, administrators were reading papers and talking with other people. There is even a case where the ES failed and the administrator failed to notice, which will be shown later.

At LOA 5 and 6 there is a division of the workload between the ES and the administrator. As shown in Figure 4.1 the division of labor for LOA 5 is 75% on the ES and 25% on the administrator. The administrator is dealing with attacks of impact four and lower, which were mailbox overflow attempts and internal Denial of Service (DOS) alerts. Neither of these attacks is critical to the network and can be dealt with or ignored without major consequences to the network.

At LOA 6 there are very few attacks that have an impact of 7 so most of the workload is on the administrator; 99.655% is on the administrator while less than 1% is on the ES.

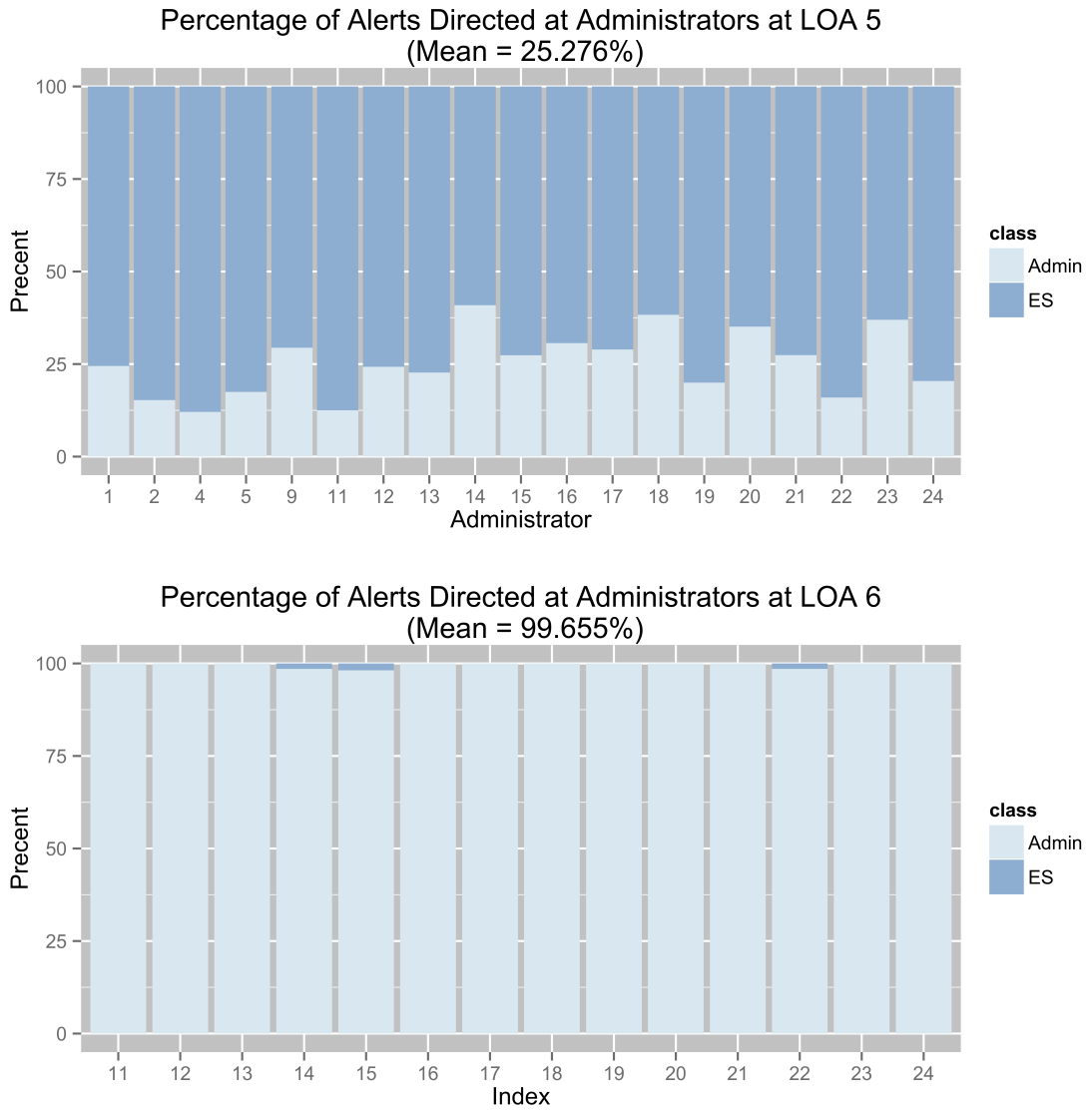

Figure 4.1: Percentage of Alerts Directed at Administrator Across Both Platforms

As shown in Figure 4.2 the workload is balanced across both of the platforms. The same system to generate the workload was used in the execution of all test cases.

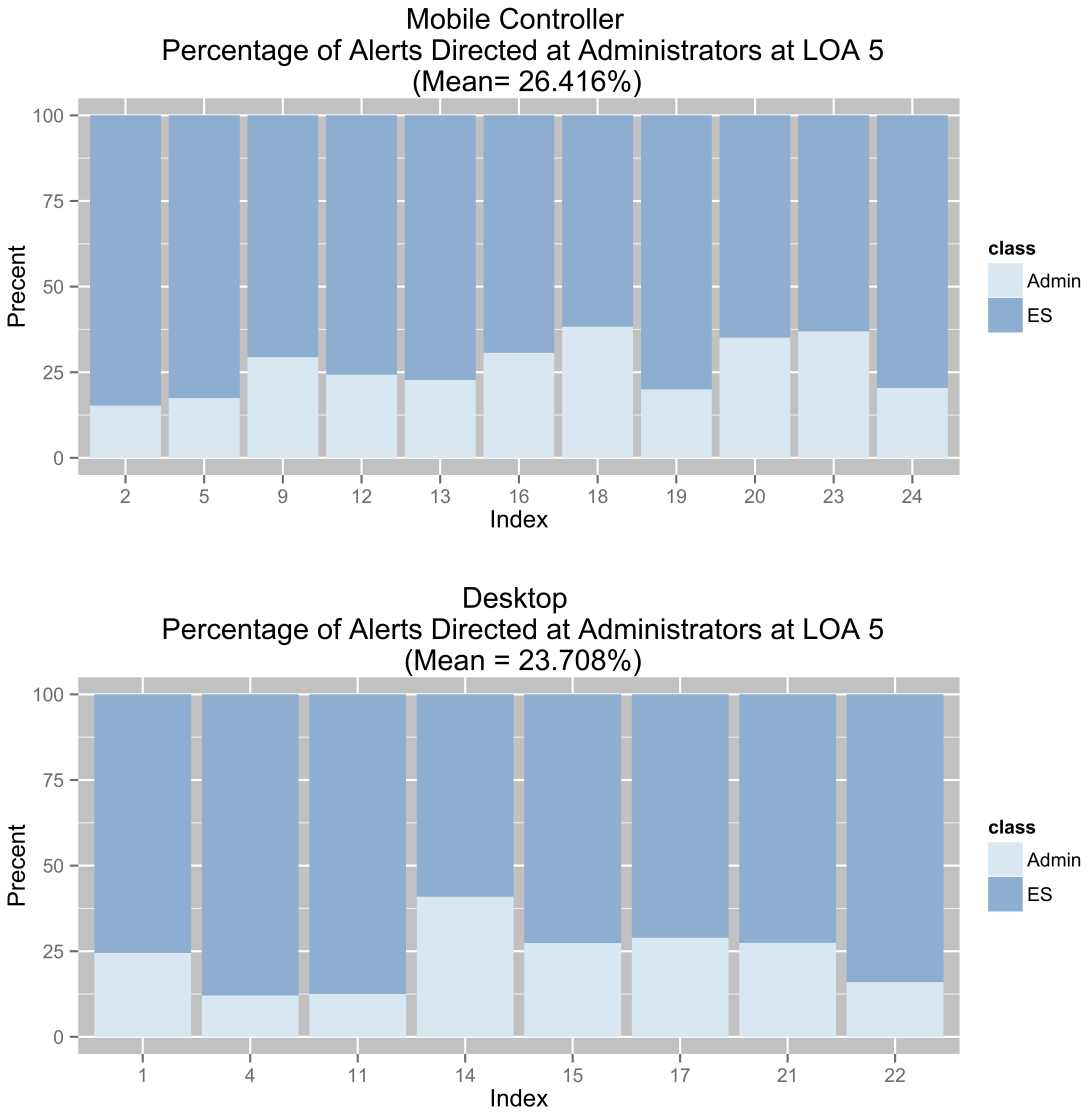

Figure 4.2: Percentage of Alerts Directed at Administrator for LOA 5

Additionally, at LOA 6 [Figure 4.3] the workload is balanced across both of the platforms. In this distribution at LOA 6 the MNDI received no alerts that

were automated, making it the same as the fully manual LOA 8. The difference in automation between the MNDI and the desktop is still within 1%.

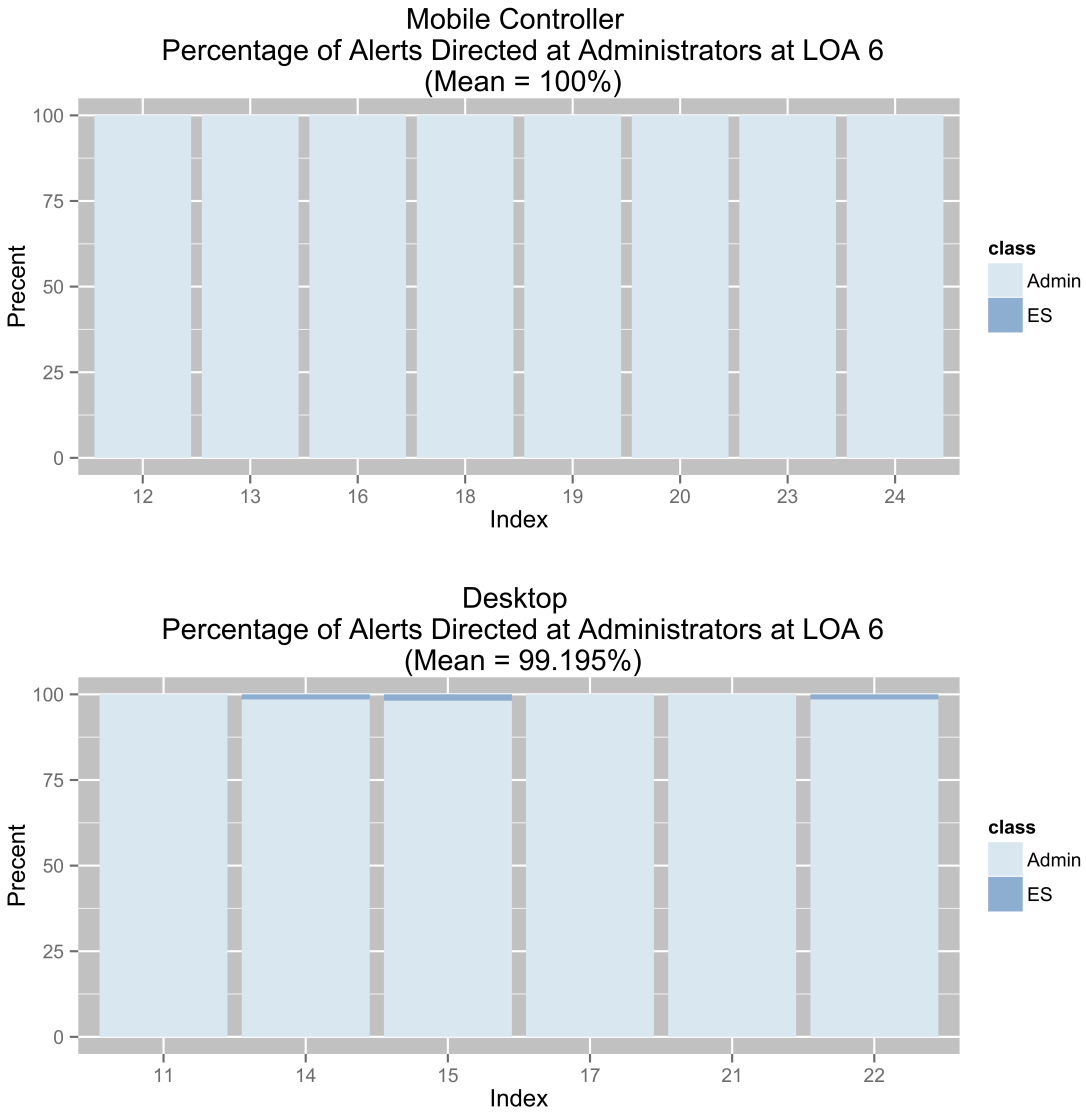

Figure 4.3: Percentage of Alerts Directed at Administrator for LOA 5

4.2 System Efficiency and Administrator SA

This section discusses the overall effectiveness of the administrator, the ES system and how that changes between the two platforms with the various LOAs and how the administrator's SA is affected. It is shown that there is very little interaction at full automation and the administrator is overwhelmed at the lower LOAs. There is a large performance gap between the two platforms and that affects the conclusion as to what LOA is the most effective.

4.2.1 MNDI.

The MNDI is a very effective platform when compared to the desktop. Even at the lower LOAs the administrators resolved most of the alerts [Table 4.1] where the desktop was consistently lower except for LOA 1 [Table 4.3] due to a failure on the MNDI; without this outlier 99.295% of the alerts were resolved.

Table 4.1: MNDI: Percent Resolved and Administrators Additional Actions

{Outlier in LOA 1 removed}

LOA	**Percent**	**Admin Additional Actions**
LOA 1	99.296%	Admin #23 : 21 actions
LOA 5	96.44%	Admin #18 : 2 actions Admin #24 : 1 action
LOA 6	68.528%	Admin #13 : 3 actions Admin #23 : 3 Actions
LOA 8	67.027%	Admin #19 : 1 actions

At LOA 1 all of the alerts are resolved except for administrators 9 and 19[Figure 4.4]. Administrator 9 only had one alert unresolved and that is most likely

a alert that came in just as the test case was concluding. For administrator 19, there was a failure in the ES and they failed to identify it. This caused 67.5% of the alerts to go unresolved. This demonstrates automation bias and lack of SA on the network [Figure 4.5][Table 4.1].

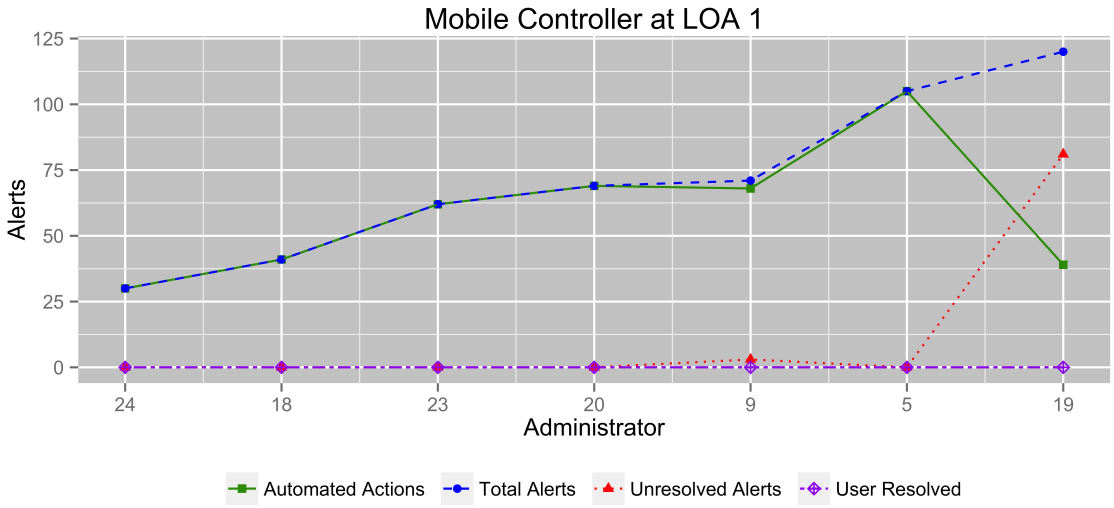

Figure 4.4: MNDI: Alerts Resolved for LOA 1

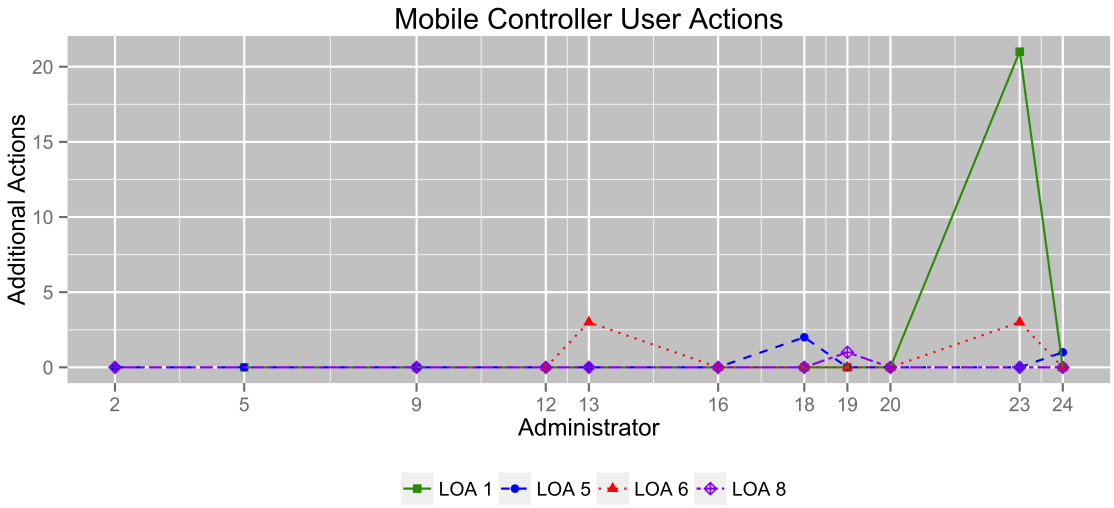

Figure 4.5: MNDI: Administrators Additional Actions Performed

At LOA 5, there were two administrators that performed additional actions [Figure 4.6][Table 4.1]. Furthermore the administrators were much more involved in the network operations because they had to resolve approximately 25% of the alerts. This indicates an increase of SA because of that involvement. LOA 5 also is shown to be a well-balanced workload for those administrators because a mean of 96.44% [Table 4.1]of the alerts were resolved, indicating that the administrators were not overwhelmed.

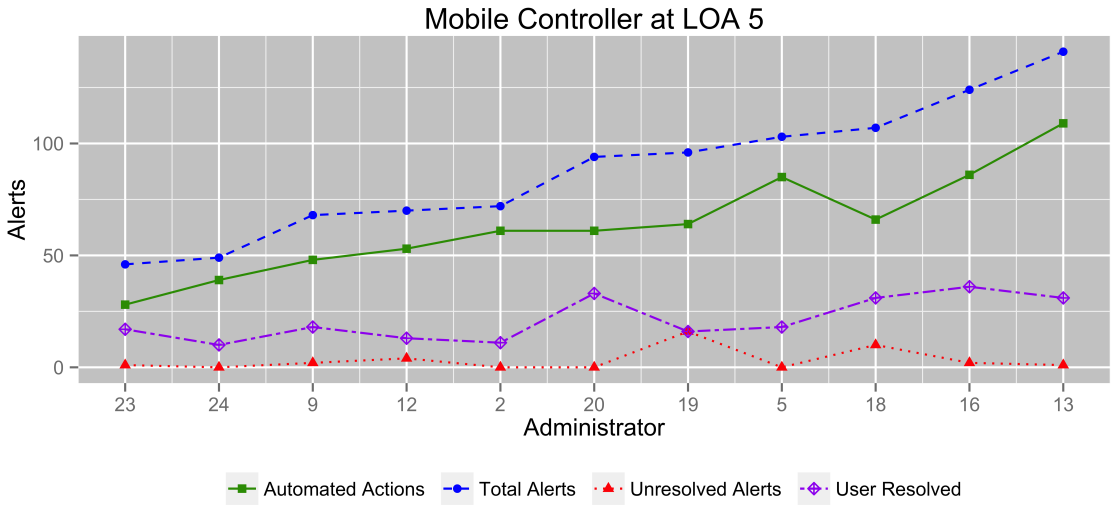

Figure 4.6: MNDI: Alerts Resolved for LOA 5

LOAs 6 and 8 were identical in the fact that there were no alerts automated by the ES and both only had a mean of 67% of the alerts resolved [Table 4.1]. With LOAs 6 and 8 both being fully manual we can see where the administrator starts to get overwhelmed on the MNDI, which is around when the number of alerts reach between seventy-five and one-hundred [Figure 4.7]. The reason there were no automated alerts at LOA 6 is there is only one alert at a level seven and it was never detected by the IDS during the MNDI test cases.

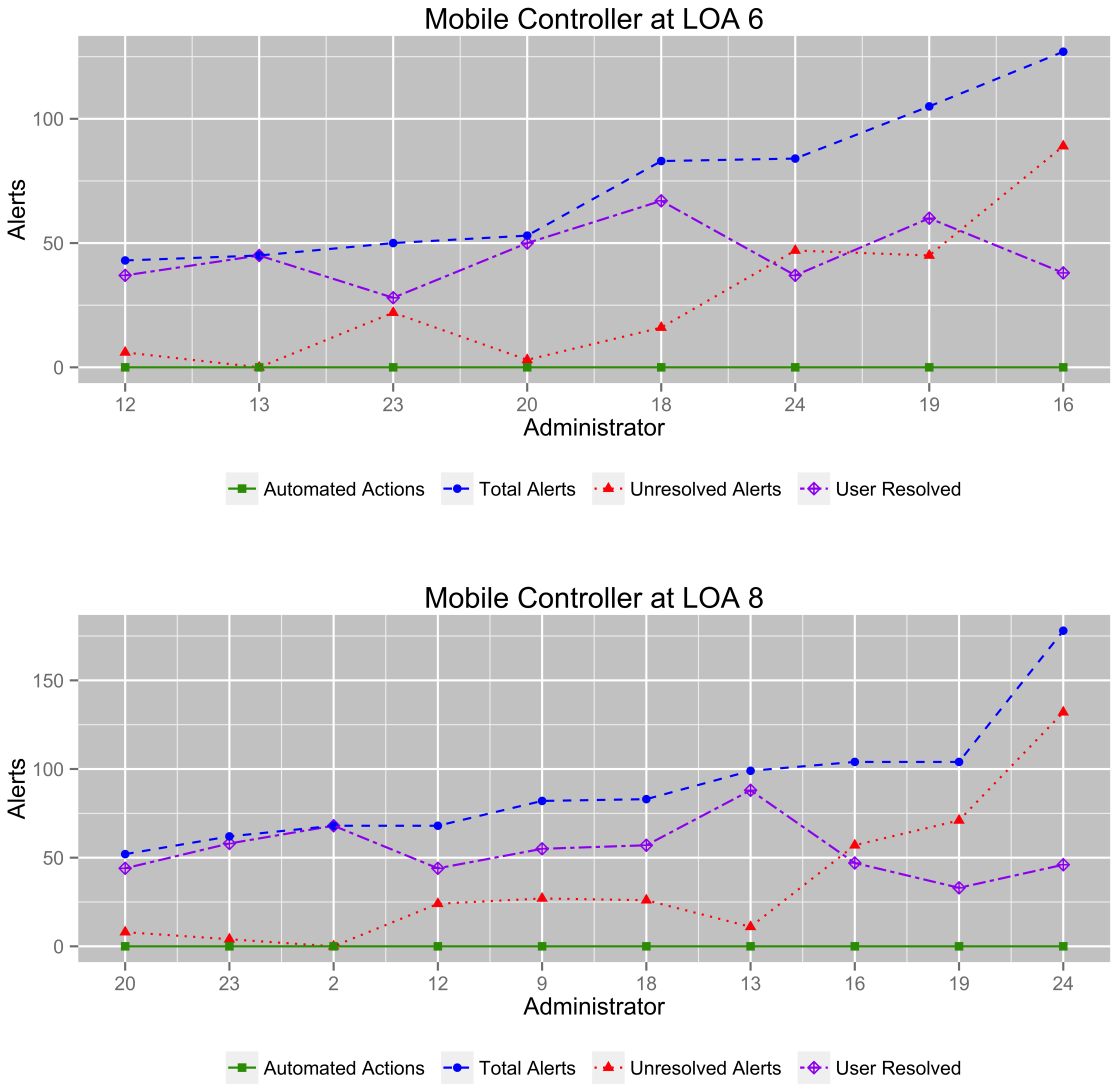

Figure 4.7: MNDI: Alerts Resolved for LOA 6 and 8

As illustrated in the boxplot [Figure 4.8] and confirmed with a Wilcoxon Rank Sum Test [Table 4.2], there is no statistical difference between LOA 1 and 5 or between LOA 6 and 8. In LOA 6 and 8 we see the administrator getting overwhelmed as the number of incoming alerts increases thus those LOAs decrease in effectiveness. We see comparable performance at At LOA 1 and 5. LOA 1 will lead to greater automation

bias and skill degradation, therefore LOA 5 is the best balance between manual and automation for the MNDI.

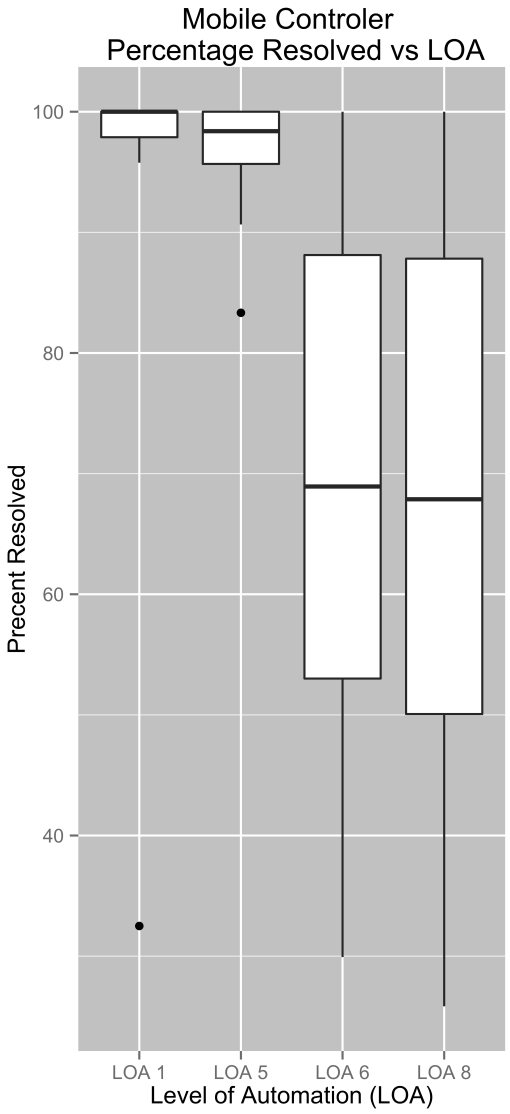

Figure 4.8: MNDI: Difference in Performance by LOA

Table 4.2: MNDI: Wilcoxon Rank Sum Test H_0 = Location Shift Equals 0

LOA vs LOA	**P-value**	**Reject H_0**
LOA 1 vs 5	0.3839	No
LOA 1 vs 6	0.04216	Yes
LOA 1 vs 8	0.02177	Yes
LOA 5 vs 6	0.01106	Yes
LOA 5 vs 8	0.003266	Yes
LOA 6 vs 8	1	No

4.2.2 Desktop.

The desktop, with the exception of LOA 1, is a less effective platform for our test cases. At LOA 1 the ES resolved all the alerts and the same ES is used in both platforms. We only see a 5% difference from the MNDI to the desktop, discounting the outlier in the MNDI data set. However, we see a significant difference in effectiveness at LOAs 5, 6, and 8 across the platforms [Table 4.1][Table 4.3]. There is a 23%, 46%, and 47% drop respectively between platforms at the same LOAs.

There is also a greater difference between the LOAs on the desktop then on the MNDI. There is a 21% drop in alerts resolved between LOA 1 and 5 and a 52% drop between LOA 5 and LOAs 6 and 8 where there is a 3% and 28% difference on the MNDI. The desktop platform did elicit more additional administrator actions, because of a lower automation bias. There is a distinct difference between the number of additional actions performed and LOA settings, LOA 5 had the highest level of user action followed by 6, 8 and 1 [Table 4.3][Figure 4.9].

Table 4.3: Desktop: Percent Resolved and Administrators Additional Actions

LOA	**Percent**	**Admin Additional Actions**
LOA 1	94.436%	Admin #21 : 3 actions
LOA 5	73.791%	Admin #11 : 6 actions Admin #14 : 1 action Admin #15 : 1 action Admin #21 : 4 actions Admin #22 : 18 actions
LOA 6	22.249%	Admin #11 : 4 actions Admin #14 : 3 actions Admin #22 : 7 actions
LOA 8	19.826%	Admin #17 : 1 action Admin #22 : 11 actions

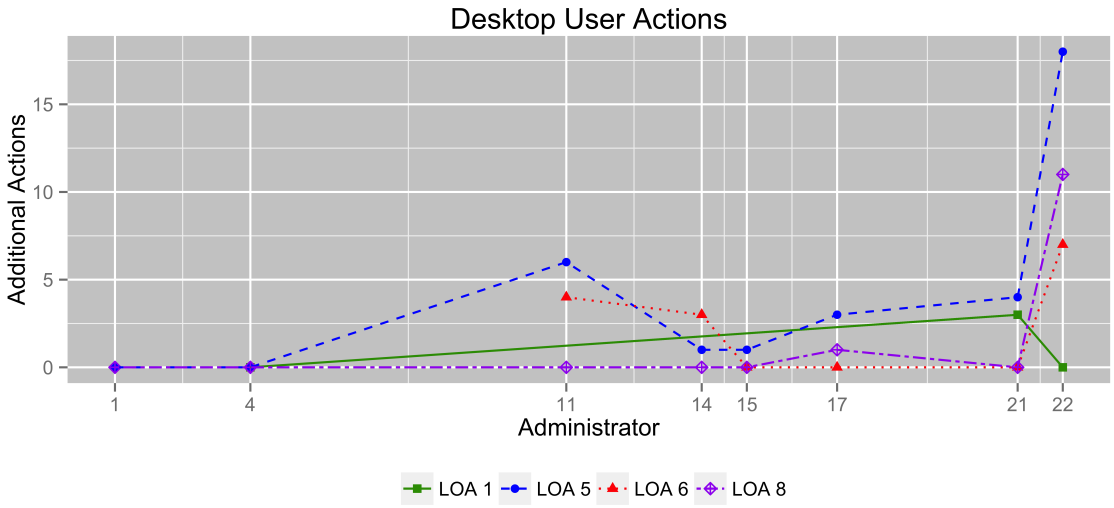

Figure 4.9: Desktop: Administrators Additional Actions Performed

There are two factors that could have significantly attributed to the large difference between LOA 1 and 5 [Figure 4.10]. The desktop interface can be confusing because the administrator is required to correlate 3 different sets of information to gain a clear picture of the network. It might take longer for the administrator to gain SA on network operations. The only way to determine that is to run the study again with a longer test case run time.

At LOA 1 we see only one administrator performing additional action where at LOA 5 there are 5 administrators performing additional actions. This shows that administrators were more active on the network at LOA 5, showing a reduced automation bias, reducing the human out-of-the-loop performance problem and skill degradation [Table 4.3][Figure 4.10][Figure 4.9]. A 74% resolution rate is too low to be acceptable for defending a network. If we extrapolate the workload at a LOA of 4, we get a 15% workload instead of the 25% we see at LOA 5. This leads to the conclusion that the best LOA is platform dependent.

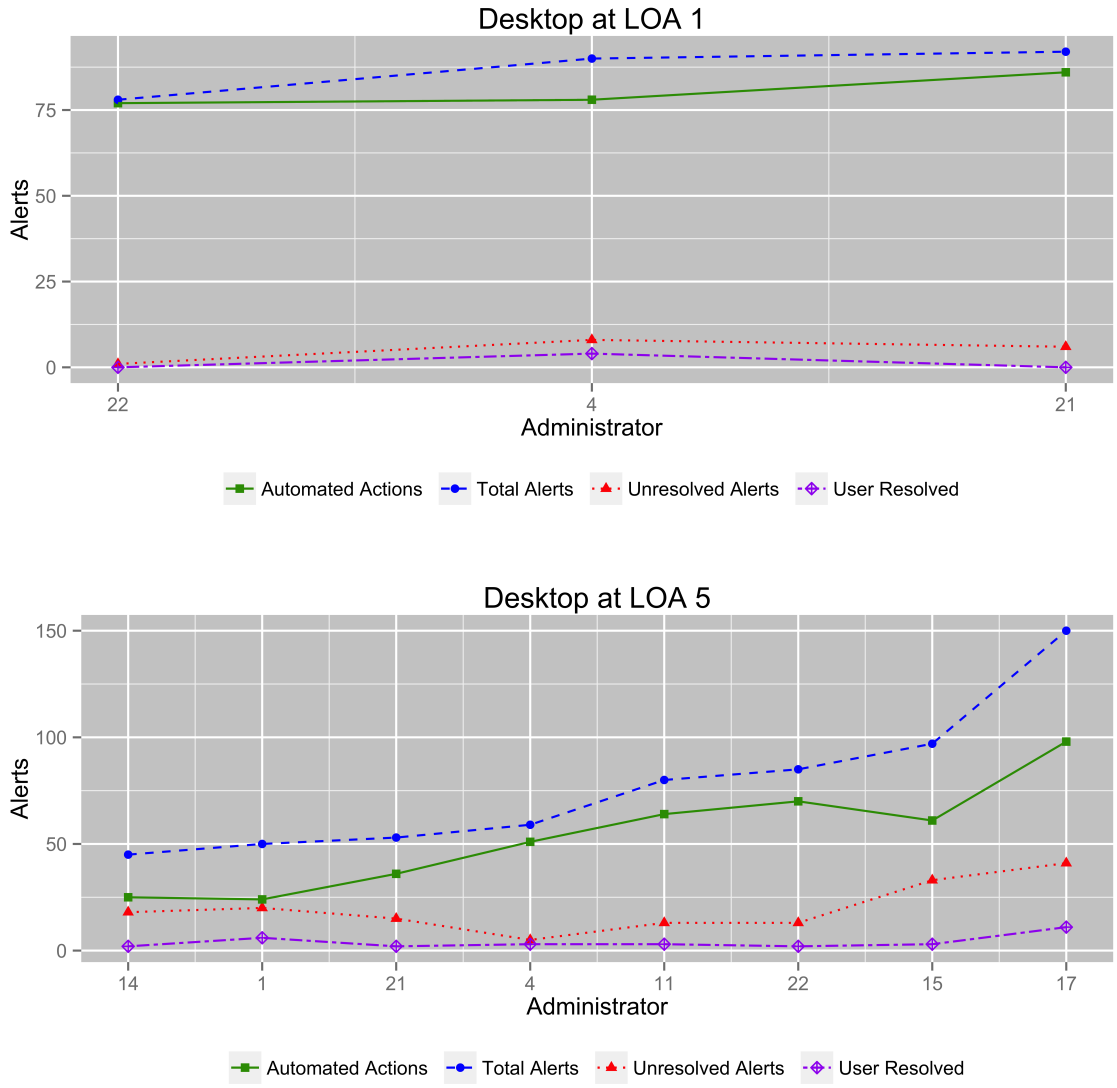

Figure 4.10: Desktop: Alerts Resolved for LOA 1 and 5

At LOA 6 and 8 the administrator is overwhelmed immediately, even when the alert level is just over forty-five. Just like the during the MNDI tests, LOAs 6 and 8 are statically the same [Figure 4.11][Figure 4.12][Table 4.4] and well below an acceptable efficiency levels.

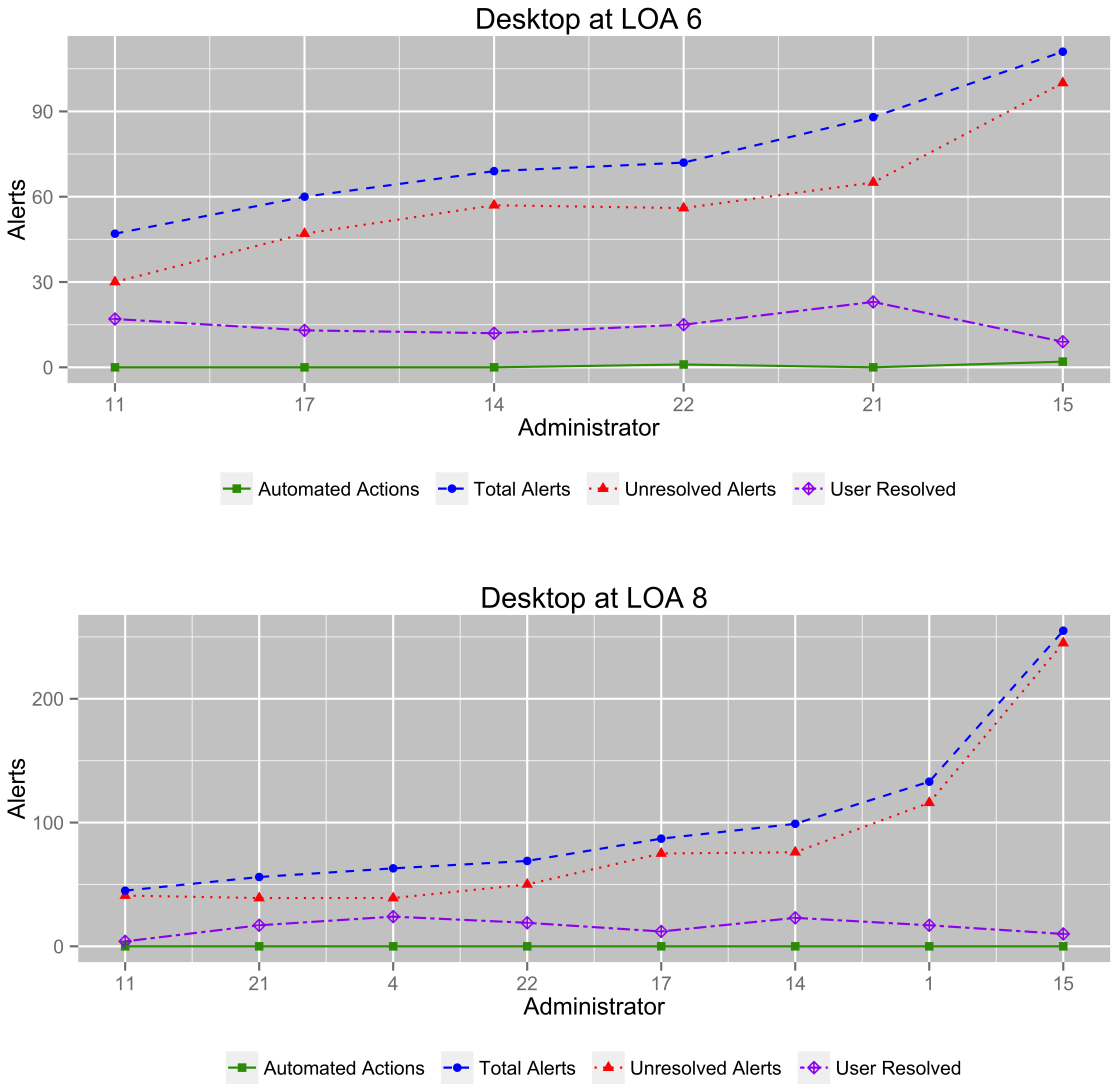

Figure 4.11: Desktop: Alerts Resolved for LOA 6 and 8

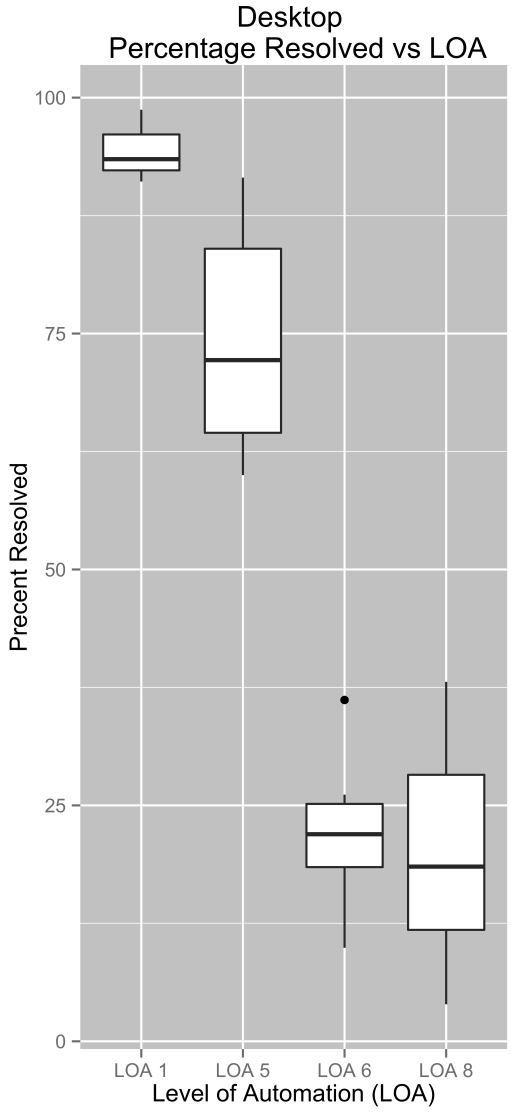

Figure 4.12: Desktop: Difference in Performance by LOA

Table 4.4: Desktop: Wilcoxon Rank Sum Test H_0 L̄ocation Shift Equals 0

LOA vs LOA	**P-value**	**Reject** H_0
LOA 1 vs 5	0.0315	Yes
LOA 1 vs 6	0.02381	Yes
LOA 1 vs 8	0.01212	Yes
LOA 5 vs 6	0.002388	Yes
LOA 5 vs 8	0.000931	Yes
LOA 6 vs 8	0.8518	No

V. Conclusion

This research looked at the performance of the administrator with the ES and the administrators SA throughout the experiment at varying LOA settings, workloads, and among 2 different network interface platforms. The goal is to determine at what LOA the performance and the SA of the administrator are preserved and if the optimal LOA setting is platform dependent.

5.1 MNDI

The administrator could handle a workload of 25% and still perform at the same level as when the alerts were fully automated. When the administrator's workload was 100%, they were not overwhelmed until the alert count reached seventy-five to one hundred. LOA 5 was shown to be the optimal level to keep balance between performance and SA. The administrators were actively engaged in network operations, preserving their SA and skills and reducing the out-of-the-loop performance issue and automation bias.

The MNDI did introduce a higher level of automation bias than the desktop even at the lower LOAs due to the recommended actions the ES produced for all manual alerts. This was also evident in the number of additional actions. The administrators responded to the MNDIs generated alerts very quickly and efficiently but did not initiate a significant number of additional administrator actions.

5.2 Desktop

The 25% workload was overwhelming for the administrator even at low alert counts. If the LOA was set to level 4 the workload would have been reduced to 15% and might have shown a more acceptable resolution percentage. Even at a 25% workload the administrators demonstrated a higher level of SA, shown by the

number of additional actions. The automation bias was reduced because of the lack of recommended actions as well as there was less skill degradation because the desktop interface is more representative of the standard network system interface where the MNDI is a unique interface to itself.

5.3 Conclusion

For the MNDI, LOA 5 was the optimal level of automation to preserve performance and SA but it introduced a higher level of automation bias than the desktop platform. LOA 5 on the desktop presented to great a workload for the administrator to handle. LOA 4, which would have had a workload of about 15% and might have been a more optimal choice. LOA 5 still out-performed the other LOA setting and was better at preserving the administrators SA. The desktop also reduced the automation bias as compared with the MNDI.

5.4 Future work

After experimenting with LOA settings at various workloads, and platforms it is worth exploring if dynamic LOA setting would further improve performance and SA. Implementing an artificial intelligence system to control the LOA would enhance the systems ability to keep the administrator at optimal performance even with extreme shifts in workload or changing interface platforms.

The interface platforms used in this research are representative of the new MNDI capabilities and a open-source simple desktop interface. future work should expand the scope of the network and focus on platforms that are used within DoD to show that these conclusions hold for more complex desktop interfaces and where LOA might prove an asset to the military mission.

Bibliography

[1] Basic analysis and security engine (BASE) project. http://base.secureideas.net, January 2013.

[2] PBNJ - network scanner. http://pbnj.sourceforge.net, January 2013.

[3] SNORT - IDS/IPS. http://www.snort.org, January 2013.

[4] Spiceworks - network monitoring. http://www.spiceworks.com/, January 2013.

[5] United States Air Force Chief Scientist (AF/ST). Technology horizons : A vision for air force science & technology during 2010-2030, May 2010.

[6] J. Elin Bahner, Anke-Dorothea Hper, and Dietrich Manzey. Misuse of automated decision aids: Complacency, automation bias and the impact of training experience. *International Journal of Human-Computer Studies*, 66(9):688–699, September 2008.

[7] Matt Bishop. *Computer Security: Art and Science*. Addison-Wesley Professional. Part of the Pearson Custom Library: Computer Science series., 1st edition, December 2002.

[8] Thomas R. Carretta and Guy A. French. Combating vigilance decrements in a sustained attention task: Lack of support for the utility of a cognitive intervention secondary task. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 56(1):1446–1450, September 2012.

[9] Erci Cole, Ronald Krutz, and James Conley. *Network Security Bible*. Wiley Publishing, Inc., second edition, 2009.

[10] M.L. Cummings. Automation bias in intelligent time critical decision support systems. In *AIAA 1st Intelligent Systems Technical Conference*, pages 1–6, 2004.

[11] Mica R. Endsley and Esin O. Kiris. The out-of-the-loop performance problem and level of control in automation. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(2):381–394, June 1995.

[12] M.R. Endsley. Situation awareness: The development and application of a theoretical framework. In *Proceedings of the Human Factors Society 37th Annual Meeting Santa Monica*, pages 39–40, Sanata Monica, CA, 1993.

[13] M.R. Endsley, E. Onal, and D.B. Kaber. The impact of intermediate levels of automation on situation awareness and performance in dynamic control systems. In *, Proceedings of the 1997 IEEE Sixth Conference on Human Factors and Power Plants, 1997. 'Global Perspectives of Human Factors in Power Generation'*, pages 7/7 –712, June 1997.

[14] D.I. Gertman. Human factors and data fusion as part of control systems resilience. In *Human System Interactions, 2009. HSI '09. 2nd Conference on*, pages 642 –647, May 2009.

[15] James Hannan. *Mobile Command and Control For Cyber Defense and Situational Awareness*. PhD thesis, Air Force Institute of Technology, 2013.

[16] Michael Howard, David LeBlanc, and John Viega. *24 Deadly Sins of Software Security: Programming Flaws and How to Fix Them*. McGraw-Hill Osborne Media, 1 edition, September 2009.

[17] R.E.T. Jones, E.S. Connors, and M.R. Endsley. A framework for representing agent and human situation awareness. In *Cognitive Methods in Situation*

Awareness and Decision Support (CogSIMA), 2011 IEEE First International Multi-Disciplinary Conference on, pages 226 –233, February 2011.

[18] David Kaber and Mica Endsley. The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task. *heoretical Issues in Ergonomics Science*, 5(2):113–153, 1999.

[19] D.B. Kaber and M.R. Endsley. Out-of-the-loop performance problems and the use of itermediate levels of automation for improved control system functioning and safety. *Process Safety Progress*, 16(3):126–131, 1997.

[20] David Kennedy, Jim O'Gorman, Devon Kearns, and Mati Aharoni. *Metasploit The Penetration Tester's Guide*. No Starch Press, Inc., 2011.

[21] B Mcguinness and L Foy. A subjective measure of SA: the crew awareness rating scale (CARS). 2000.

[22] Robert Molloy and Raja Parasuraman. Monitoring an automated system for a single failure: Vigilance and task complexity effects. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 38(2):311–322, June 1996.

[23] R. Parasuraman, S. Galster, P. Squire, H. Furukawa, and C. Miller. A flexible delegation-type interface enhances system performance in human supervision of multiple robots: empirical studies with RoboFlag. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 35(4):481 – 493, July 2005.

[24] R. Parasuraman, T.B. Sheridan, and C.D. Wickens. A model for types and levels of human interaction with automation. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 30(3):286 –297, May 2000.

[25] Raja Parasuraman. Designing automation for human use: empirical studies and quantitative models. *Ergonomics*, 43(7):931–951, 2000.

[26] Raja Parasuraman and Dietrich H. Manzey. Complacency and bias in human use of automation: An attentional integration. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 52(3):381–410, June 2010.

[27] Evan Raulerson. *Modeling Cyber Situational Awareness through Data Fusion*. PhD thesis, Air Force Institute of Technology, 2013.

[28] Karen Scarfone and Peter Mell. Guide to intrusion detection and prevention systems (IDPS), February 2007.

[29] LINDA J. SKITKA, KATHLEEN L. MOSIER, and MARK BURDICK. Does automation bias decision-making? *International Journal of Human-Computer Studies*, 51(5):991–1006, November 1999.

[30] E. Tyugu. Artificial intelligence in cyber defense. In *Cyber Conflict (ICCC), 2011 3rd International Conference on*, pages 1 –11, June 2011.

[31] Joel S. Warm, Raja Parasuraman, and Gerald Matthews. Vigilance requires hard mental work and is stressful. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 50(3):433–441, June 2008.

[32] Melanie Wright and David Kaber. Effects of automation of information-processing functions on teamwork. *Human Factors*, January 2005.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
21–03–2013	Master's Thesis	Oct 2011–Mar 2013

4. TITLE AND SUBTITLE

Cognitive Augmentation for Network Defense

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Emge, James E., Captain, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB, OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-13-M-16

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

ATTN: Richard Fedors
25 Electronic Parkway, Rome, NY 13441

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/RISC

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A.
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Traditionally, when a task is considered for automation it is a binary decision, either the task was completely automated or it remains manual. LOA is a departure from the tradition use of automation in cyber defense. When a task is automated, it removes the human administrator from the performance of the task, compromising their SA of the state of the network. When the administrator loses SA of the network performance and its current state, failure recovery time becomes much longer. This is because the administrators must orient themselves to the current state of the network at the time of failure and determine the cause of the failure before repairs or supplemental operations can occur. LOA attempts to mitigate this problem by keeping the administrator engaged in network tasks along side the automation agent. Keeping the administrator aware of both the automated system's performance and the performance of the network, while taking advantage of the automation system's speed and the complex decision making of the administrator. This research applies LOA to computer network defense during cyber attacks. The goal is to find the most efficient LOA that keeps the administrator engaged in the defense of the network while preserving efficiency. The LOA allows the administrator to supplement and/or correct the automated system, while the automated system handles the time sensitive events to keep the administrator from being overwhelmed or the network from being compromised.

15. SUBJECT TERMS

Cyber Defense, Level of Automation, LOA, Mobile Network Controller, Mobile, Out of the loop, automation bias, skill degradation, Situational Awareness

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Kenneth Hopkinson (ENG)
U	U	U	UU	63	19b. TELEPHONE NUMBER *(include area code)* 937-255-3636x4579 Kenneth.Hopkinson@afit.edu